

EXERCÍCIO PRÁTICO 06

ALUNO: GUILHERME GOMES DE BRITES

MATRÍCULA: 808721

Parte1 - Responda

1. O que é um arquivo fonte?
 A. um arquivo de texto que contém instruções de linguagem de programação.
B. um subdiretório que contém os programas.
C. um arquivo que contém dados para um programa.
D. um documento que contém os requisitos para um projeto.

2. O que é um registrador?
A. parte do sistema de computador que mantém o controle dos parâmetros do sistema.
B. uma parte do processador que possui um padrão de bits.
 C. parte do processador que contém o seu número de série único.
D. parte do bus de sistema que contém dados.

3. Qual o caractere que, na linguagem assembly do SPIM, inicia um comentário?
 A. #
B. \$
C. //
D. *

4. Quantos bits há em cada instrução de máquina MIPS?
A. 8
B. 16
 C. 32
D. instruções diferentes possuem diferentes comprimentos.

5. O que é o contador de programa?
A. um registrador que mantém a conta do número de erros durante a execução de um programa.
B. uma parte do processador que contém o endereço da primeira palavra de dados.
C. uma variável na montadora que os números das linhas do arquivo de origem.
 D. parte do processador que contém o endereço da próxima instrução de máquina para ser obtida.

6. Ao executarmos uma instrução, quanto será adicionado ao contador de programa?
A. 1
B. 2
 C. 4
D. 8

7. O que é uma diretiva, tal como a diretiva .text?
 A. uma instrução em linguagem assembly que resulta em uma instrução em linguagem de máquina.
B. uma das opções de menu do sistema SPIM.
C. uma instrução em linguagem de máquina que faz com que uma operação sobre os dados ocorra.
D. uma declaração que diz o montador algo sobre o que o programador quer, mas não corresponde diretamente a uma instrução de máquina.

8. O que é um endereço simbólico?
A. um local de memória que contém dados simbólicos.
B. um byte na memória que contém o endereço de dados.
C. símbolo dado como argumento para uma directiva.
 D. um nome usado no código-fonte em linguagem assembly para um local na memória.

9. Em qual endereço o simulador SPIM coloca a primeira instrução de máquina quando ele está sendo executado?
 A. 0x00000000
B. 0x00400000
C. 0x10000000
D. 0xFFFFFFFF
10. Algumas instruções de máquina possuem uma constante como um dos operandos. Como é chamado tal operando?
 A. operando imediato
B. operando embutido
C. operando binário
D. operando de máquina
11. Como é chamada uma operação lógica executada entre bits de cada coluna dos operandos para produzir um bit de resultado para cada coluna?
 A. operação lógica
 B. operação bitwise
C. operação binária
D. operação coluna
12. Quando uma operação é de fato executada, como estão os operandos na ALU?
A. Pelo menos um operando deve ser de 32 bit.
B. Cada operando pode ser de qualquer tamanho.
C. Ambos operandos devem que vir de registros.
 D. Cada um dos registradores deve possuir 32 bit.
13. Dezesseis bits de dados de uma instrução de ori são usados como um operando imediato. Durante execução, o que deve ser feito primeiro?
A. Os dados são estendidos em zero à direita por 16 bits.
B. Os dados são estendidos em zero à esquerda por 16 bits.
 C. Nada precisa ser feito.
D. Apenas 16 bits são usados pelo outro operando.
14. Qual das instruções seguintes armazenam no registrador \$5 um padrão de bits que representa positivo 48?
A. ori \$5,\$0,0x48
B. ori \$5,\$5,0x48
 C. ori \$5,\$0,48
D. ori \$0,\$5,0x48
15. A instrução de ori pode armazenar o complemento de dois de um número em um registrador?
A. Não.
 B. Sim.
16. Qual das instruções seguintes limpa todos os bits no registrador \$8 com exceção do byte de baixa ordem que fica inalterado?
A. ori \$8,\$8,0xFF
B. ori \$8,\$0,0x00FF
 C. xori \$8,\$8,0xFF
D. andi \$8,\$8,0xFF
17. Qual é o resultado de um ou exclusivo de padrão sobre ele mesmo?

17. Qual é o resultado de um ou exclusivo de padrão sobre ele mesmo?

- A. Todos os bits em zero.
- B. Todos os bits em um.
- C. O padrão original utilizado.
- D. O resultado é o contrário do original.

18. Todas as instruções de máquina têm os mesmos campos?

- A. Não. Diferentes de instruções de máquina possuem campos diferentes.
- B. Não. Cada instrução de máquina é completamente diferente de qualquer outra.
- C. Sim. Todas as instruções de máquina têm os mesmos campos na mesma ordem.
- D. Sim. Todas as instruções de máquina têm os mesmos campos, mas eles podem estar em ordens diferentes.

Parte2 - Implementar em MIPS/MARS os seguintes programas (usando apenas as instruções indicadas)

//programa 1

```
mips1.asm*
1 ori $s0, $zero, 2 # a = 2
2 ori $s1, $zero, 3 # b = 3
3 ori $s2, $zero, 4 # c = 4
4 ori $s3, $zero, 5 # d = 5
5
6 add $t0, $s0, $s1 # t0 = a + b
7 add $t1, $s2, $s3 # t1 = c + d
8
9 sub $s4, $t0, $t1 # x = t0 - t1
10
11 sub $t2, $s0, $s1 # t2 = a - b
12 add $s5, $t2, $s4 # y = t2 + x
13
14 sub $s1, $s4, $s5 # b = x - y
```

//programa 2

```
programa1.asm programa2.asm programa3.asm
1 ori $s0, $zero, 1 # x = 1
2
3 add $t0, $t0, $s0 # x
4 add $t0, $t0, $s0 # 2x
5 add $t0, $t0, $s0 # 3x
6 add $t0, $t0, $s0 # 4x
7 add $t0, $t0, $s0 # 5x
8
9 addi $s1, $t0, 15
10
```

//programa 3

programa1.asm	programa2.asm	programa3.asm	programa1.asm	programa2.asm	programa3.asm
1 ori \$s0, \$zero, 3 # x = 3			28 add \$t1, \$t1, \$s1 # 11y		
2 ori \$s1, \$zero, 4 # y = 4			29 add \$t1, \$t1, \$s1 # 12y		
3			30 add \$t1, \$t1, \$s1 # 13y		
4 add \$t0, \$s0, \$s0 # 2x			31 add \$t1, \$t1, \$s1 # 14y		
5 add \$t0, \$t0, \$s0 # 3x			32 add \$t1, \$t1, \$s1 # 15y		
6 add \$t0, \$t0, \$s0 # 4x			33 add \$t1, \$t1, \$s1 # 16y		
7 add \$t0, \$t0, \$s0 # 5x			34 add \$t1, \$t1, \$s1 # 17y		
8 add \$t0, \$t0, \$s0 # 6x			35 add \$t1, \$t1, \$s1 # 18y		
9 add \$t0, \$t0, \$s0 # 7x			36 add \$t1, \$t1, \$s1 # 19y		
10 add \$t0, \$t0, \$s0 # 8x			37 add \$t1, \$t1, \$s1 # 20y		
11 add \$t0, \$t0, \$s0 # 9x			38 add \$t1, \$t1, \$s1 # 21y		
12 add \$t0, \$t0, \$s0 # 10x			39 add \$t1, \$t1, \$s1 # 22y		
13 add \$t0, \$t0, \$s0 # 11x			40 add \$t1, \$t1, \$s1 # 23y		
14 add \$t0, \$t0, \$s0 # 12x			41 add \$t1, \$t1, \$s1 # 24y		
15 add \$t0, \$t0, \$s0 # 13x			42 add \$t1, \$t1, \$s1 # 25y		
16 add \$t0, \$t0, \$s0 # 14x			43 add \$t1, \$t1, \$s1 # 26y		
17 add \$t0, \$t0, \$s0 # 15x			44 add \$t1, \$t1, \$s1 # 27y		
18			45 add \$t1, \$t1, \$s1 # 28y		
19 add \$t1, \$s1, \$s1 # 2y			46 add \$t1, \$t1, \$s1 # 29y		
20 add \$t1, \$t1, \$s1 # 3y			47 add \$t1, \$t1, \$s1 # 30y		
21 add \$t1, \$t1, \$s1 # 4y			48 add \$t1, \$t1, \$s1 # 31y		
22 add \$t1, \$t1, \$s1 # 5y			49 add \$t1, \$t1, \$s1 # 32y		
23 add \$t1, \$t1, \$s1 # 6y			50 add \$t1, \$t1, \$s1 # 33y		
24 add \$t1, \$t1, \$s1 # 7y			51 add \$t1, \$t1, \$s1 # 34y		
25 add \$t1, \$t1, \$s1 # 8y			52 add \$t1, \$t1, \$s1 # 35y		
26 add \$t1, \$t1, \$s1 # 9y			53 add \$t1, \$t1, \$s1 # 36y		
27 add \$t1, \$t1, \$s1 # 10y			54 add \$t1, \$t1, \$s1 # 37y		

Text Segment						Data Segment						
Bit#	Address	Code	Basic	Source						Name	Number	Value
	0x04000000	0x34100003	ori \$t0, \$zero, 3 # x = 3	1: ori \$s0, \$zero, 3 # x = 3						zzero	0	0
	0x04000000	0x34110004	ori \$t1, \$zero, 4	2: ori \$s1, \$zero, 4 # y = 4						zat	1	0
	0x04000008	0x02104020	add \$s0,\$t0,\$t0	4: add \$t0, \$t0, \$s0 # 2x						cvl	2	0
	0x04000008	0x02104020	add \$s0,\$t0,\$t0	5: add \$t0, \$t0, \$s0 # 3x						czl	3	0
	0x04000012	0x02104020	add \$s0,\$t0,\$t0	6: add \$t0, \$t0, \$s0 # 4x						ca0	4	0
	0x04000012	0x02104020	add \$s0,\$t0,\$t0	7: add \$t0, \$t0, \$s0 # 5x						ca1	5	0
	0x04000016	0x02104020	add \$s0,\$t0,\$t0	8: add \$t0, \$t0, \$s0 # 6x						ca2	6	0
	0x04000016	0x02104020	add \$s0,\$t0,\$t0	9: add \$t0, \$t0, \$s0 # 7x						ca3	7	0
	0x04000020	0x02104020	add \$s0,\$t0,\$t0	10: add \$t0, \$t0, \$s0 # 8x						ca4	8	0
	0x04000020	0x02104020	add \$s0,\$t0,\$t0	11: add \$t0, \$t0, \$s0 # 9x						ct1	9	249
	0x04000024	0x02104020	add \$s0,\$t0,\$t0	12: add \$t0, \$t0, \$s0 # 10x						ct2	10	0
	0x04000024	0x02104020	add \$s0,\$t0,\$t0	13: add \$t0, \$t0, \$s0 # 11x						ct3	11	317
	0x04000028	0x02104020	add \$s0,\$t0,\$t0	14: add \$t0, \$t0, \$s0 # 12x						ct4	12	0
	0x04000028	0x02104020	add \$s0,\$t0,\$t0	15: add \$t0, \$t0, \$s0 # 13x						ct5	13	0
	0x04000032	0x02104020	add \$s0,\$t0,\$t0	16: add \$t0, \$t0, \$s0 # 14x						ct6	14	0
	0x04000032	0x02104020	add \$s0,\$t0,\$t0	17: add \$t0, \$t0, \$s0 # 15x						ct7	15	0
	0x04000036	0x02104020	add \$s0,\$t0,\$t0	18: add \$t0, \$t0, \$s0 # 16x						cz0	16	3
	0x04000036	0x02104020	add \$s0,\$t0,\$t0	19: add \$t0, \$t0, \$s0 # 17x						cz1	17	4
	0x04000040	0x02104020	add \$s0,\$t0,\$t0	20: add \$t0, \$t0, \$s0 # 18x						cz2	18	0
	0x04000040	0x02104020	add \$s0,\$t0,\$t0	21: add \$t0, \$t0, \$s0 # 19x						cz3	19	0
	0x04000044	0x02104020	add \$s0,\$t0,\$t0	22: add \$t0, \$t0, \$s0 # 20x						cz4	20	0
	0x04000044	0x02104020	add \$s0,\$t0,\$t0	23: add \$t0, \$t0, \$s0 # 21x						cz5	21	0
	0x04000048	0x02104020	add \$s0,\$t0,\$t0	24: add \$t0, \$t0, \$s0 # 22x						cz6	22	0
	0x04000048	0x02104020	add \$s0,\$t0,\$t0	25: add \$t0, \$t0, \$s0 # 23x						cz7	23	0
	0x04000052	0x02104020	add \$s0,\$t0,\$t0	26: add \$t0, \$t0, \$s0 # 24x						cz8	24	0
	0x04000052	0x02104020	add \$s0,\$t0,\$t0	27: add \$t0, \$t0, \$s0 # 25x						cz9	25	0
	0x04000056	0x02104020	add \$s0,\$t0,\$t0	28: add \$t0, \$t0, \$s0 # 26x						cz0	26	0
	0x04000056	0x02104020	add \$s0,\$t0,\$t0	29: add \$t0, \$t0, \$s0 # 27x						cz1	27	0
	0x04000060	0x02104020	add \$s0,\$t0,\$t0	30: add \$t0, \$t0, \$s0 # 28x						czp	28	26468224
	0x04000060	0x02104020	add \$s0,\$t0,\$t0	31: add \$t0, \$t0, \$s0 # 29x						czp	29	214745460
	0x04000064	0x02104020	add \$s0,\$t0,\$t0	32: add \$t0, \$t0, \$s0 # 30x						czra	31	0
	0x04000064	0x02104020	add \$s0,\$t0,\$t0	33: add \$t0, \$t0, \$s0 # 31x						ps	32	4194640
	0x04000068	0x02104020	add \$s0,\$t0,\$t0	34: add \$t0, \$t0, \$s0 # 32x						hi	35	0
	0x04000068	0x02104020	add \$s0,\$t0,\$t0	35: add \$t0, \$t0, \$s0 # 33x						lo	36	0

programa1.asm	programa2.asm	programa
---------------	---------------	----------

```

55 add $t1, $t1, $s1 # 38y
56 add $t1, $t1, $s1 # 39y
57 add $t1, $t1, $s1 # 40y
58 add $t1, $t1, $s1 # 41y
59 add $t1, $t1, $s1 # 42y
60 add $t1, $t1, $s1 # 43y
61 add $t1, $t1, $s1 # 44y
62 add $t1, $t1, $s1 # 45y
63 add $t1, $t1, $s1 # 46y
64 add $t1, $t1, $s1 # 47y
65 add $t1, $t1, $s1 # 48y
66 add $t1, $t1, $s1 # 49y
67 add $t1, $t1, $s1 # 50y
68 add $t1, $t1, $s1 # 51y
69 add $t1, $t1, $s1 # 52y
70 add $t1, $t1, $s1 # 53y
71 add $t1, $t1, $s1 # 54y
72 add $t1, $t1, $s1 # 55y
73 add $t1, $t1, $s1 # 56y
74 add $t1, $t1, $s1 # 57y
75 add $t1, $t1, $s1 # 58y
76 add $t1, $t1, $s1 # 59y
77 add $t1, $t1, $s1 # 60y
78 add $t1, $t1, $s1 # 61y
79 add $t1, $t1, $s1 # 62y
80 add $t1, $t1, $s1 # 63y
81 add $t1, $t1, $s1 # 64y

```

82 add \$t1, \$t1, \$s1 # 65y

```

83 add $t1, $t1, $s1 # 66y
84 add $t1, $t1, $s1 # 67y
85
86 add $t3, $t0, $t1 # j = 15*x + 67*y
87 addi $t3, $t3, 4 # 4*j

```

//programa 4

programa1.asm	programa2.asm	programa3.asm*
---------------	---------------	----------------

```

1 ori $s0, $zero, 3 # x = 3
2 ori $s1, $zero, 4 # y = 4
3
4 sll $t0, $s0, 4 # t0 = 16x
5 sub $t0, $t0, $s0 # t0 = 15x
6
7 sll $t1, $s1, 6 # t1 = 64y
8 sll $t2, $s1, 1 # t2 = 2y
9
10 add $t2, $t2, $s1 # t2 = 3y
11 add $t1, $t1, $t2 # t1 = 67y
12
13 add $t0, $t0, $t1 # t0 = 15x + 67y
14 sll $s2, $t0, 2 # z = 4 * t0
15

```

//programa 5

```
programa1.asm  programa2.asm  program

1 ori $t0, $zero, 50000
2 sll $s0, $t0, 1
3
4 ori $t1, $zero, 50000
5 sll $s1, $t1, 2
6
7 add $s2, $s0, $s1
8
```

The screenshot shows the Immunity Debugger interface with two main panes: the Text Segment and the Data Segment.

Text Segment:

Blkt	Address	Code	Basic	Source
	0x00400000	0x400c3500	ori \$t0, \$zero, \$0000	
	0x00400004	0x00890400	sll \$t1, \$t0, 1	
	0x00400008	0x400c3500	ori \$t2, \$zero, \$0000	
	0x0040000c	0x00890400	sll \$t3, \$t2, 1	
	0x00400010	0x400c3500	ori \$t4, \$zero, \$0000	
	0x00400014	0x00890400	sll \$t5, \$t4, 1	
	0x00400018	0x400c3500	ori \$t6, \$zero, \$0000	
	0x0040001c	0x00890400	sll \$t7, \$t6, 1	
	0x00400020	0x400c3500	ori \$t8, \$zero, \$0000	
	0x00400024	0x00890400	sll \$t9, \$t8, 1	
	0x00400028	0x400c3500	ori \$t10, \$zero, \$0000	
	0x0040002c	0x00890400	sll \$t11, \$t10, 1	
	0x00400030	0x400c3500	ori \$t12, \$zero, \$0000	
	0x00400034	0x00890400	sll \$t13, \$t12, 1	
	0x00400038	0x400c3500	ori \$t14, \$zero, \$0000	
	0x00400042	0x00890400	sll \$t15, \$t14, 1	
	0x00400046	0x400c3500	ori \$t16, \$zero, \$0000	
	0x00400050	0x00890400	sll \$t17, \$t16, 1	
	0x00400054	0x400c3500	ori \$t18, \$zero, \$0000	
	0x00400058	0x00890400	sll \$t19, \$t18, 1	
	0x00400062	0x400c3500	ori \$t20, \$zero, \$0000	
	0x00400066	0x00890400	sll \$t21, \$t20, 1	
	0x00400070	0x400c3500	ori \$t22, \$zero, \$0000	
	0x00400074	0x00890400	sll \$t23, \$t22, 1	
	0x00400078	0x400c3500	ori \$t24, \$zero, \$0000	
	0x00400082	0x00890400	sll \$t25, \$t24, 1	
	0x00400086	0x400c3500	ori \$t26, \$zero, \$0000	
	0x00400090	0x00890400	sll \$t27, \$t26, 1	
	0x00400094	0x400c3500	ori \$t28, \$zero, \$0000	
	0x00400098	0x00890400	sll \$t29, \$t28, 1	
	0x004000a2	0x400c3500	ori \$t30, \$zero, \$0000	
	0x004000a6	0x00890400	sll \$t31, \$t30, 1	
	0x004000b0	0x400c3500	ori \$t32, \$zero, \$0000	
	0x004000b4	0x00890400	sll \$t33, \$t32, 1	
	0x004000b8	0x400c3500	ori \$t34, \$zero, \$0000	
	0x004000bc	0x00890400	sll \$t35, \$t34, 1	
	0x004000c0	0x400c3500	ori \$t36, \$zero, \$0000	
	0x004000c4	0x00890400	sll \$t37, \$t36, 1	
	0x004000c8	0x400c3500	ori \$t38, \$zero, \$0000	
	0x004000d2	0x00890400	sll \$t39, \$t38, 1	
	0x004000d6	0x400c3500	ori \$t40, \$zero, \$0000	
	0x004000e0	0x00890400	sll \$t41, \$t40, 1	
	0x004000e4	0x400c3500	ori \$t42, \$zero, \$0000	
	0x004000e8	0x00890400	sll \$t43, \$t42, 1	
	0x004000f2	0x400c3500	ori \$t44, \$zero, \$0000	
	0x004000f6	0x00890400	sll \$t45, \$t44, 1	
	0x004000f0	0x400c3500	ori \$t46, \$zero, \$0000	
	0x004000f4	0x00890400	sll \$t47, \$t46, 1	
	0x004000f8	0x400c3500	ori \$t48, \$zero, \$0000	
	0x004000fc	0x00890400	sll \$t49, \$t48, 1	
	0x00400100	0x400c3500	ori \$t50, \$zero, \$0000	
	0x00400104	0x00890400	sll \$t51, \$t50, 1	
	0x00400108	0x400c3500	ori \$t52, \$zero, \$0000	
	0x0040010c	0x00890400	sll \$t53, \$t52, 1	
	0x00400110	0x400c3500	ori \$t54, \$zero, \$0000	
	0x00400114	0x00890400	sll \$t55, \$t54, 1	
	0x00400118	0x400c3500	ori \$t56, \$zero, \$0000	
	0x0040011c	0x00890400	sll \$t57, \$t56, 1	
	0x00400120	0x400c3500	ori \$t58, \$zero, \$0000	
	0x00400124	0x00890400	sll \$t59, \$t58, 1	
	0x00400128	0x400c3500	ori \$t60, \$zero, \$0000	
	0x0040012c	0x00890400	sll \$t61, \$t60, 1	
	0x00400130	0x400c3500	ori \$t62, \$zero, \$0000	
	0x00400134	0x00890400	sll \$t63, \$t62, 1	
	0x00400138	0x400c3500	ori \$t64, \$zero, \$0000	
	0x0040013c	0x00890400	sll \$t65, \$t64, 1	
	0x00400140	0x400c3500	ori \$t66, \$zero, \$0000	
	0x00400144	0x00890400	sll \$t67, \$t66, 1	
	0x00400148	0x400c3500	ori \$t68, \$zero, \$0000	
	0x0040014c	0x00890400	sll \$t69, \$t68, 1	
	0x00400150	0x400c3500	ori \$t70, \$zero, \$0000	
	0x00400154	0x00890400	sll \$t71, \$t70, 1	
	0x00400158	0x400c3500	ori \$t72, \$zero, \$0000	
	0x0040015c	0x00890400	sll \$t73, \$t72, 1	
	0x00400160	0x400c3500	ori \$t74, \$zero, \$0000	
	0x00400164	0x00890400	sll \$t75, \$t74, 1	
	0x00400168	0x400c3500	ori \$t76, \$zero, \$0000	
	0x0040016c	0x00890400	sll \$t77, \$t76, 1	
	0x00400170	0x400c3500	ori \$t78, \$zero, \$0000	
	0x00400174	0x00890400	sll \$t79, \$t78, 1	
	0x00400178	0x400c3500	ori \$t80, \$zero, \$0000	
	0x0040017c	0x00890400	sll \$t81, \$t80, 1	
	0x00400180	0x400c3500	ori \$t82, \$zero, \$0000	
	0x00400184	0x00890400	sll \$t83, \$t82, 1	
	0x00400188	0x400c3500	ori \$t84, \$zero, \$0000	
	0x0040018c	0x00890400	sll \$t85, \$t84, 1	
	0x00400190	0x400c3500	ori \$t86, \$zero, \$0000	
	0x00400194	0x00890400	sll \$t87, \$t86, 1	
	0x00400198	0x400c3500	ori \$t88, \$zero, \$0000	
	0x0040019c	0x00890400	sll \$t89, \$t88, 1	
	0x004001a0	0x400c3500	ori \$t90, \$zero, \$0000	
	0x004001a4	0x00890400	sll \$t91, \$t90, 1	
	0x004001a8	0x400c3500	ori \$t92, \$zero, \$0000	
	0x004001ac	0x00890400	sll \$t93, \$t92, 1	
	0x004001b0	0x400c3500	ori \$t94, \$zero, \$0000	
	0x004001b4	0x00890400	sll \$t95, \$t94, 1	
	0x004001b8	0x400c3500	ori \$t96, \$zero, \$0000	
	0x004001bc	0x00890400	sll \$t97, \$t96, 1	
	0x004001c0	0x400c3500	ori \$t98, \$zero, \$0000	
	0x004001c4	0x00890400	sll \$t99, \$t98, 1	
	0x004001c8	0x400c3500	ori \$t100, \$zero, \$0000	
	0x004001cc	0x00890400	sll \$t101, \$t100, 1	
	0x004001d0	0x400c3500	ori \$t102, \$zero, \$0000	
	0x004001d4	0x00890400	sll \$t103, \$t102, 1	
	0x004001d8	0x400c3500	ori \$t104, \$zero, \$0000	
	0x004001dc	0x00890400	sll \$t105, \$t104, 1	
	0x004001e0	0x400c3500	ori \$t106, \$zero, \$0000	
	0x004001e4	0x00890400	sll \$t107, \$t106, 1	
	0x004001e8	0x400c3500	ori \$t108, \$zero, \$0000	
	0x004001ec	0x00890400	sll \$t109, \$t108, 1	
	0x004001f0	0x400c3500	ori \$t110, \$zero, \$0000	
	0x004001f4	0x00890400	sll \$t111, \$t110, 1	
	0x004001f8	0x400c3500	ori \$t112, \$zero, \$0000	
	0x004001fc	0x00890400	sll \$t113, \$t112, 1	
	0x00400200	0x400c3500	ori \$t114, \$zero, \$0000	
	0x00400204	0x00890400	sll \$t115, \$t114, 1	
	0x00400208	0x400c3500	ori \$t116, \$zero, \$0000	
	0x0040020c	0x00890400	sll \$t117, \$t116, 1	
	0x00400210	0x400c3500	ori \$t118, \$zero, \$0000	
	0x00400214	0x00890400	sll \$t119, \$t118, 1	
	0x00400218	0x400c3500	ori \$t120, \$zero, \$0000	
	0x0040021c	0x00890400	sll \$t121, \$t120, 1	
	0x00400220	0x400c3500	ori \$t122, \$zero, \$0000	
	0x00400224	0x00890400	sll \$t123, \$t122, 1	
	0x00400228	0x400c3500	ori \$t124, \$zero, \$0000	
	0x0040022c	0x00890400	sll \$t125, \$t124, 1	
	0x00400230	0x400c3500	ori \$t126, \$zero, \$0000	
	0x00400234	0x00890400	sll \$t127, \$t126, 1	
	0x00400238	0x400c3500	ori \$t128, \$zero, \$0000	
	0x0040023c	0x00890400	sll \$t129, \$t128, 1	
	0x00400240	0x400c3500	ori \$t130, \$zero, \$0000	
	0x00400244	0x00890400	sll \$t131, \$t130, 1	
	0x00400248	0x400c3500	ori \$t132, \$zero, \$0000	
	0x0040024c	0x00890400	sll \$t133, \$t132, 1	
	0x00400250	0x400c3500	ori \$t134, \$zero, \$0000	
	0x00400254	0x00890400	sll \$t135, \$t134, 1	
	0x00400258	0x400c3500	ori \$t136, \$zero, \$0000	
	0x0040025c	0x00890400	sll \$t137, \$t136, 1	
	0x00400260	0x400c3500	ori \$t138, \$zero, \$0000	
	0x00400264	0x00890400	sll \$t139, \$t138, 1	
	0x00400268	0x400c3500	ori \$t140, \$zero, \$0000	
	0x0040026c	0x00890400	sll \$t141, \$t140, 1	
	0x00400270	0x400c3500	ori \$t142, \$zero, \$0000	
	0x00400274	0x400c3500	sll \$t143, \$t142, 1	
	0x00400278	0x400c3500	ori \$t144, \$zero, \$0000	
	0x0040027c	0x400c3500	sll \$t145, \$t144, 1	
	0x00400280	0x400c3500	ori \$t146, \$zero, \$0000	
	0x00400284	0x400c3500	sll \$t147, \$t146, 1	
	0x00400288	0x400c3500	ori \$t148, \$zero, \$0000	
	0x0040028c	0x400c3500	sll \$t149, \$t148, 1	
	0x00400290	0x400c3500	ori \$t150, \$zero, \$0000	
	0x00400294	0x400c3500	sll \$t151, \$t150, 1	
	0x00400298	0x400c3500	ori \$t152, \$zero, \$0000	
	0x0040029c	0x400c3500	sll \$t153, \$t152, 1	
	0x004002a0	0x400c3500	ori \$t154, \$zero, \$0000	
	0x004002a4	0x400c3500	sll \$t155, \$t154, 1	
	0x004002a8	0x400c3500	ori \$t156, \$zero, \$0000	
	0x004002ac	0x400c3500	sll \$t157, \$t156, 1	
	0x004002b0	0x400c3500	ori \$t158, \$zero, \$0000	
	0x004002b4	0x400c3500	sll \$t159, \$t158, 1	
	0x004002b8	0x400c3500	ori \$t160, \$zero, \$0000	
	0x004002bc	0x400c3500	sll \$t161, \$t160, 1	
	0x004002c0	0x400c3500	ori \$t162, \$zero, \$0000	
	0x004002c4	0x400c3500	sll \$t163, \$t162, 1	
	0x004002c8	0x400c3500	ori \$t164, \$zero, \$0000	
	0x004002cc	0x400c3500	sll \$t165, \$t164, 1	
	0x004002d0	0x400c3500	ori \$t166, \$zero, \$0000	
	0x004002d4	0x400c3500	sll \$t167, \$t166, 1	
	0x004002d8	0x400c3500	ori \$t168, \$zero, \$0000	
	0x004002dc	0x400c3500	sll \$t169, \$t168, 1	
	0x004002e0	0x400c3500	ori \$t170, \$zero, \$0000	
	0x004002e4	0x400c3500	sll \$t171, \$t170, 1	
	0x004002e8	0x400c3500	ori \$t172, \$zero, \$0000	
	0x004002ec	0x400c3500	sll \$t173, \$t172, 1	
	0x004002f0	0x400c3500	ori \$t174, \$zero, \$0000	
	0x004002f4	0x400c3500	sll \$t175, \$t174, 1	
	0x004002f8	0x400c3500	ori \$t176, \$zero, \$0000	
	0x004002fc	0x400c3500	sll \$t177, \$t176, 1	
	0x00400300	0x400c3500	ori \$t178, \$zero, \$0000	
	0x00400304	0x400c3500	sll \$t179, \$t178, 1	
	0x00400308	0x400c3500	ori \$t180, \$zero, \$0000	
	0x0040030c	0x400c3500	sll \$t181, \$t180, 1	
	0x00400310	0x400c3500	ori \$t182, \$zero, \$0000	
	0x00400314	0x400c3500	sll \$t183, \$t182, 1	
	0x00400318	0x400c3500	ori \$t184, \$zero, \$0000	
	0x0040031c	0x400c3500	sll \$t185, \$t184, 1	
	0x00400320	0x400c3500	ori \$t186, \$zero, \$0000	
	0x00400324	0x400c3500	sll \$t187, \$t186, 1	
	0x00400328	0x400c3500	ori \$t188, \$zero, \$0000	
	0x0040032c	0x400c3500	sll \$t189, \$t188, 1	
	0x00400330	0x400c3500	ori \$t190, \$zero, \$0000	
	0x00400334	0x400c3500	sll \$t191, \$t190, 1	
	0x00400338	0x400c3500	ori \$t192, \$zero, \$0000	
	0x0040033c	0x400c3500	sll \$t193, \$t192, 1	
	0x00400340	0x400c3500	ori \$t194, \$zero, \$0000	
	0x00400344	0x400c3500	sll \$t195, \$t194, 1	
	0x00400348	0x400c3500	ori \$t196, \$zero, \$0000	
	0x0040034c			

//programa 6

The screenshot shows a debugger interface with multiple tabs at the top: "programa1.asm", "programa2.asm", "programa3.asm", and "programa4.asm". The "programa2.asm" tab is selected. The assembly code in the editor is:

```
1 ori $t0, $zero, 0x7FFF
2 sll $t0, $t0, 16
3 ori $s0, $t0, 0xFFFF
4
5 ori $t1, $zero, 37500
6 sll $s1, $t1, 3 # s1 = 300.000
7
8 sll $t2, $s1, 2 # t2 = 1.200.000
9 sub $s2, $s0, $t2
10
```

The screenshot shows two windows below the assembly editor. The left window is titled "Text Segment" and displays a memory dump of the text section. The right window is titled "Registers" and shows the processor register state.

Register	Name	Number	Value
gzero	gzero	0	0
fat	fat	1	0
cr0	cr0	2	0
cr1	cr1	3	0
ab0	ab0	4	0
ab2	ab2	5	0
ab1	ab1	6	0
ab3	ab3	7	0
cr0	cr0	8	2147483112
cr1	cr1	9	37500
cr2	cr2	10	1200000
cr3	cr3	11	0
cr4	cr4	12	0
cr5	cr5	13	0
cr6	cr6	14	0
cr7	cr7	15	0
cr0	cr0	16	2147483647
g1	g1	17	300000
g2	g2	18	2146283647
g3	g3	19	0
g4	g4	20	0
g5	g5	21	0
g6	g6	22	0
g7	g7	23	0
g8	g8	24	0
g9	g9	25	0
g10	g10	26	0
g11	g11	27	0
gp	gp	28	284488224
sp	sp	29	2147479543
fp	fp	30	0
ra	ra	31	0
pc	pc		419432
hi	hi		0
lo	lo		0

//programa 7

The screenshot shows a debugger interface with multiple tabs at the top: "programa1.asm", "programa2.asm", "programa3.asm", and "programa4.asm". The "programa2.asm" tab is selected. The assembly code in the editor is:

```
1 ori $8, $0, 0x01
2
3 sll $t0, $t0, 31
4 sra $t0, $t0, 31
5
```

Text Segment				Labels					
Blgt	Address	Code	Basic	Source					
	0x01400000	0x40000001 ori \$8, \$0, 0x00000001		l: ori \$8, \$0, 0x01					
	0x01400004	0x40047cf sll \$8, \$8, 0x0000001f		b: sll \$8, \$0, 31					
	0x01400008	0x400847c sra \$8, \$8, 0x0000001f		d: sra \$8, \$0, 31					

Data Segment		Labels						
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10100000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10100020	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10100040	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10100060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10100080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x101000a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x101000c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x101000e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10100100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10100120	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10100140	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10100160	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10101300	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

//programa 8

programa1.asm	programa2.asm	programa3.asm
1 ori \$8, \$zero, 0x1234		
2 sll \$8, \$8, 16		
3 ori \$8, \$8, 0x5678		
4 # t0 = 0x12345678		
5		
6 srl \$9, \$8, 24		
7 # t1 = 0x000000012		
8		
9 sll \$10, \$8, 8		
10 srl \$10, \$10, 24		
11 # t2 = 0x000000034		
12		
13 sll \$11, \$8, 16		
14 srl \$11, \$11, 24		
15 # t3 = 0x000000034		
16		
17 sll \$12, \$8, 24		
18 srl \$12, \$12, 24		
19 # t4 = 0x000000078		
20		

Text Segment			Labels			Registers		
Byte	Address	Code	Basic	Source		Name	Number	Value
	0x00400000	0x40001234	ori \$t0,\$0,0x00001234	1: ori \$t0,\$0,0x00001234		\$zero	0	0x00000000
	0x00400004	0x70004400	lui \$t0,0x00000010	2: lui \$t0,0x00000010		\$at	1	0x00000001
	0x00400008	0x30005670	ori \$t0,\$t0,0x00005670	3: ori \$t0,\$t0,0x00005670		\$v0	2	0x00000000
	0x00400010	0x70004400	lui \$t1,0x00000010	4: lui \$t1,0x00000010		\$v1	3	0x00000000
	0x00400012	0x30005670	ori \$t1,\$t1,0x00005670	5: ori \$t1,\$t1,0x00005670		\$a0	4	0x00000000
	0x00400014	0x70004400	lui \$t2,0x00000010	6: lui \$t2,0x00000010		\$a1	5	0x00000000
	0x00400016	0x30005670	ori \$t2,\$t2,0x00005670	7: ori \$t2,\$t2,0x00005670		\$a2	6	0x00000000
	0x00400018	0x70004400	lui \$t3,0x00000010	8: lui \$t3,0x00000010		\$a3	7	0x00000000
	0x0040001c	0x30005670	ori \$t3,\$t3,0x00005670	9: ori \$t3,\$t3,0x00005670		\$t0	8	0x12345678
	0x00400020	0x30004400	lui \$t4,0x00000010	10: lui \$t4,0x00000010		\$t1	9	0x00000012
	0x00400024	0x30004400	lui \$t5,0x00000010	11: lui \$t5,0x00000010		\$t2	10	0x00000034
	0x00400028	0x30004400	lui \$t6,0x00000010	12: lui \$t6,0x00000010		\$t3	11	0x00000056
	0x0040002c	0x30004400	lui \$t7,0x00000010	13: lui \$t7,0x00000010		\$t4	12	0x00000078
	0x00400030	0x30004400	lui \$t8,0x00000010	14: lui \$t8,0x00000010		\$t5	13	0x00000090
	0x00400034	0x30004400	lui \$t9,0x00000010	15: lui \$t9,0x00000010		\$t6	14	0x00000000
	0x00400038	0x30004400	lui \$t10,0x00000010	16: lui \$t10,0x00000010		\$t7	15	0x00000000
	0x00400040	0x30004400	lui \$t11,0x00000010	17: lui \$t11,0x00000010		\$t8	16	0x00000000
	0x00400044	0x30004400	lui \$t12,0x00000010	18: lui \$t12,0x00000010		\$t9	17	0x00000000
	0x00400048	0x30004400	lui \$t13,0x00000010	19: lui \$t13,0x00000010		\$t10	18	0x00000000
	0x0040004c	0x30004400	lui \$t14,0x00000010	20: lui \$t14,0x00000010		\$t11	19	0x00000000
	0x00400050	0x30004400	lui \$t15,0x00000010	21: lui \$t15,0x00000010		\$t12	20	0x00000000
	0x00400054	0x30004400	lui \$t16,0x00000010	22: lui \$t16,0x00000010		\$t13	21	0x00000000
	0x00400058	0x30004400	lui \$t17,0x00000010	23: lui \$t17,0x00000010		\$t14	22	0x00000000
	0x00400060	0x30004400	lui \$t18,0x00000010	24: lui \$t18,0x00000010		\$t15	23	0x00000000
	0x00400064	0x30004400	lui \$t19,0x00000010	25: lui \$t19,0x00000010		\$t16	24	0x00000000
	0x00400068	0x30004400	lui \$t20,0x00000010	26: lui \$t20,0x00000010		\$t17	25	0x00000000
	0x00400070	0x30004400	lui \$t21,0x00000010	27: lui \$t21,0x00000010		\$t18	26	0x00000000
	0x00400074	0x30004400	lui \$t22,0x00000010	28: lui \$t22,0x00000010		\$t19	27	0x00000000
	0x00400078	0x30004400	lui \$t23,0x00000010	29: lui \$t23,0x00000010		\$t20	28	0x00000000
	0x00400080	0x30004400	lui \$t24,0x00000010	30: lui \$t24,0x00000010		\$t21	29	0xffffffff
	0x00400084	0x30004400	lui \$t25,0x00000010	31: lui \$t25,0x00000010		\$t22	30	0x00000000
	0x00400088	0x30004400	lui \$t26,0x00000010			\$t23	31	0x00000000

//programa 9

programa8.asm	programa9.asm*
1	.data
2	x1: .word 15
3	x2: .word 25
4	x3: .word 13
5	x4: .word 17
6	soma: .word -1
7	
8	.text
9	lui \$t0, 0x1001
10	
11	lw \$s0, 0(\$t0)
12	lw \$s1, 4(\$t0)
13	lw \$s2, 8(\$t0)
14	lw \$s3, 12(\$t0)
15	
16	add \$t1, \$s0, \$s1
17	add \$t1, \$t1, \$s2
18	add \$t1, \$t1, \$s3
19	
20	sw \$t1, 16(\$t0)
21	

Text Segment							Registers		
Bptr	Address	Code	Basic	Source			Coproc 1	Coproc 0	
	0x04000000	0xb0801001	lui \$t0, 0x1001	9: lui \$t0, 0x1001			czero	0	
	0x04000004	0xb1000000	lw \$t1, 0(\$t0)	10: lw \$t1, 0(\$t0) # x			c\$at	1	
	0x04000008	0xb1100004	lw \$t2, 4(\$t0)	11: lw \$t2, 4(\$t0) # z			c\$v0	2	
	0x0400000c	0xb1100004	lw \$t3, 8(\$t0)	12: lw \$t3, 8(\$t0)			c\$vl	3	
	0x04000010	0xb1100004	lw \$t4, 12(\$t0)	13: lw \$t4, 12(\$t0)			c\$al	4	
	0x04000014	0xb1110020	add \$t1, \$t2, \$t3	14: add \$t1, \$t2, \$t3			c\$ae	5	
	0x04000018	0xb1110020	add \$t1, \$t2, \$t4	15: add \$t1, \$t2, \$t4			c\$z0	6	
	0x0400001c	0xb1110020	add \$t1, \$t2, \$t3	16: add \$t1, \$t2, \$t3			c\$z1	7	
	0x04000020	0xb0500100	sw \$t1, 16(\$t0)	17: sw \$t1, 16(\$t0)			c\$z2	8	

Data Segment							Registers		
Address	Value (+0)	Value (+4)	Value (+8)	Value (+12)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)	
0x10010000	0x00000005	0x00000015	0x0000000d	0x00000011	0x00000046	0x00000000	0x00000000	0x00000000	
0x10010020	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x10010040	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x10010060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x10010080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x100100e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x10010100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x10010120	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x10010140	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x10010160	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	

//programa 10

programa8.asm		programa9.asm		programa10.asm				
1	.data							
2	x: .word	5						
3	z: .word	7						
4	y: .word	0	# esse valor deverá ser sobreescrito após a execução do programa.					
5								
6	.text							
7								
8	lui	\$t0,	0x1001					
9								
10	lw	\$s0,	0(\$t0) # x					
11	lw	\$s1,	4(\$t0) # z					
12								
13	sll	\$t1,	\$s0,	7 # $128x = 2^7$				
14	sub	\$t1,	\$t1,	\$s0 #127x				
15								
16	sll	\$t2,	\$s1,	6 # $64z = 2^6$				
17	add	\$t2,	\$t2,	\$s1 # 65z				
18								
19	sub	\$t3,	\$t1,	\$t2 # $127x - 65z$				
20	addi	\$t3,	\$t3,	1 # $127x - 65z + 1$				
21								
22	sw	\$t3,	8(\$t0)					
23								

Text Segment							Registers		
Bptr	Address	Code	Basic	Source			Coproc 1	Coproc 0	
	0x04000000	0xb0801001	lui \$t0, 0x1001	8: lui \$t0, 0x1001			czero	0	
	0x04000004	0xb1000000	lw \$t1, 0(\$t0)	10: lw \$t1, 0(\$t0) # x			c\$at	1	
	0x04000008	0xb1100004	lw \$t2, 4(\$t0)	11: lw \$t2, 4(\$t0) # z			c\$v0	2	
	0x0400000c	0xb1100004	lw \$t3, 8(\$t0)	12: lw \$t3, 8(\$t0)			c\$vl	3	
	0x04000010	0xb1100004	lw \$t4, 12(\$t0)	13: lw \$t4, 12(\$t0)			c\$al	4	
	0x04000014	0xb1110020	add \$t1, \$t2, \$t3	14: add \$t1, \$t2, \$t3			c\$ae	5	
	0x04000018	0xb1110020	add \$t1, \$t2, \$t4	15: add \$t1, \$t2, \$t4			c\$z0	6	
	0x0400001c	0xb1110020	add \$t1, \$t2, \$t3	16: add \$t1, \$t2, \$t3			c\$z1	7	
	0x04000020	0xb0500100	sw \$t1, 16(\$t0)	17: sw \$t1, 16(\$t0)			c\$z2	8	

Data Segment							Registers		
Address	Value (+0)	Value (+4)	Value (+8)	Value (+12)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)	
0x10010000	5	0x00000005	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	
0x10010020	0	0	0	0	0	0	0	0	
0x10010040	0	0	0	0	0	0	0	0	
0x10010060	0	0	0	0	0	0	0	0	
0x10010080	0	0	0	0	0	0	0	0	
0x100100a0	0	0	0	0	0	0	0	0	
0x100100c0	0	0	0	0	0	0	0	0	
0x100100e0	0	0	0	0	0	0	0	0	
0x10010100	0	0	0	0	0	0	0	0	
0x10010120	0	0	0	0	0	0	0	0	
0x10010140	0	0	0	0	0	0	0	0	
0x10010160	0	0	0	0	0	0	0	0	
0x10010180	0	0	0	0	0	0	0	0	
0x100101a0	0	0	0	0	0	0	0	0	
0x100101c0	0	0	0	0	0	0	0	0	

//programa 11

programa8.asm programa9.asm programa10.asm **programa11.asm*** programa7.asm programa6.asm

```

1 .data
2 x: .word 100000
3 z: .word 200000
4 y: .word 0 # esse valor deverá ser sobreescrito após a execução do programa.
5
6 .text
7
8 lui $t0, 0x1001
9
10 lw $s0, 0($t0)
11 lw $s1, 4($t0)
12
13 sub $s3, $s0, $s1
14
15 ori $t1, $zero, 37500
16 sll $s4, $t1, 3
17
18 add $s5, $s3, $s4
19
20 sw $s5, 8($t0)
21

```

Edit Execute

Text Segment			Source
Byte	Address	Code	Basic
	0x00400000	0xb0010001	lui \$t0, 0x1001
	0x00400004	0xd1100001	lw \$s0, 0(\$t0)
	0x00400008	0xd1110004	lw \$s1, 4(\$t0)
	0x0040000c	0xd1110022	sub \$s3, \$s0, \$s1
	0x00400010	0xb0395027	ori \$t1, \$zero, 37500
	0x00400014	0xc0094e0c	sll \$s4, \$t1, 3
	0x00400018	0xd0574e02	add \$s5, \$s3, \$s4
	0x0040001c	0xdax150008	sw \$s5, 8(\$t0)

Data Segment			Value (+0)	Value (+4)	Value (+8)	Value (+C)	Value (+10)	Value (+14)	Value (+18)	Value (+1C)
0x10010000	100000	200000	200000	0	0	0	0	0	0	0
0x10010020	0	0	0	0	0	0	0	0	0	0
0x10010040	0	0	0	0	0	0	0	0	0	0
0x10010060	0	0	0	0	0	0	0	0	0	0
0x10010080	0	0	0	0	0	0	0	0	0	0
0x100100a0	0	0	0	0	0	0	0	0	0	0
0x100100c0	0	0	0	0	0	0	0	0	0	0
0x100100e0	0	0	0	0	0	0	0	0	0	0
0x10010100	0	0	0	0	0	0	0	0	0	0
0x10010120	0	0	0	0	0	0	0	0	0	0
0x10010140	0	0	0	0	0	0	0	0	0	0
0x10010160	0	0	0	0	0	0	0	0	0	0
0x10010180	0	0	0	0	0	0	0	0	0	0
0x100101a0	0	0	0	0	0	0	0	0	0	0
0x100101c0	0	0	0	0	0	0	0	0	0	0

Registers	Coproc 1	Coproc 0
Name	Number	Value
rzero	0	0
\$at	1	0
\$v0	2	0
\$v1	3	0
\$a0	4	0
\$a1	5	0
\$a2	6	0
\$a3	7	0
\$t0	8	288500962
\$t1	9	37500
\$t2	10	0
\$t3	11	0
\$t4	12	0
\$t5	13	0
\$t6	14	0
\$t7	15	0
\$s0	16	100000
\$s1	17	200000
\$s2	18	0
\$s3	19	-100000
\$s4	20	300000
\$s5	21	200000
\$s6	22	0
\$s7	23	0
\$s8	24	0
\$s9	25	0
\$s0	26	0
\$t1	27	0
\$gp	28	288488274
\$sp	29	214741543
\$fp	30	0
\$ra	31	0
pc		4194306
hi		0
lo		0

0x10010000 (.data) Hexadecimal Addresses Hexadecimal Values ASCII

//programa 12

programa12.asm

```

4  pp: .word 0
5  ppp: .word 0
6
7  .text
8  lui $t0, 0x1001 # endereco do inteiro
9  addi $t1, $t0, 4
10 addi $t2, $t0, 8
11 addi $t3, $t0, 12
12
13 sw $t0, 4($t0)
14 sw $t1, 8($t0)
15 sw $t2, 12($t0)
16
17
18 lw $t1, 0($t3)
19 lw $t1, 0($t1)
20 lw $t1, 0($t1)
21 lw $t1, 0($t1)
22
23 sll $t1, $t1, 1
24
25 lw $t2, 0($t3)
26 lw $t2, 0($t2)
27 lw $t2, 0($t2)
28 sw $t1, 0($t2)
29

```

Text Segment

Blpt	Address	Code	Basic	Source
0x00400000	0x40000000	lui \$t0, 0x00001001	8: lui \$t0, 0x1001 # endereco do inteiro	
0x00400004	0x40000004	addi \$t1, \$t0, 4	9: addi \$t1, \$t0, 4	
0x00400008	0x40000008	addi \$t2, \$t0, 8	10: addi \$t2, \$t0, 8	
0x0040000c	0x4000000c	addi \$t3, \$t0, 12	11: addi \$t3, \$t0, 12	
0x00400010	0x40000010	sw \$t0, 4(\$t0)	13: sw \$t0, 4(\$t0)	
0x00400014	0x40000014	sw \$t1, 8(\$t0)	14: sw \$t1, 8(\$t0)	
0x00400018	0x40000018	sw \$t2, 12(\$t0)	15: sw \$t2, 12(\$t0)	
0x0040001c	0x4000001c	lw \$t1, 0(\$t3)	18: lw \$t1, 0(\$t3)	
0x00400020	0x40000020	lw \$t1, 0(\$t1)	19: lw \$t1, 0(\$t1)	
0x00400024	0x40000024	lw \$t1, 0(\$t1)	20: lw \$t1, 0(\$t1)	
0x00400028	0x40000028	lw \$t1, 0(\$t1)	21: lw \$t1, 0(\$t1)	
0x0040002c	0x4000002c	sll \$t1, \$t1, 1	23: sll \$t1, \$t1, 1	
0x00400030	0x40000030	lw \$t2, 0(\$t3)	25: lw \$t2, 0(\$t3)	
0x00400034	0x40000034	lw \$t2, 0(\$t2)	26: lw \$t2, 0(\$t2)	
0x00400038	0x40000038	lw \$t2, 0(\$t2)	27: lw \$t2, 0(\$t2)	
0x0040003c	0x4000003c	sw \$t1, 0(\$t2)	28: sw \$t1, 0(\$t2)	

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+12)	Value (+16)	Value (+20)	Value (+24)	Value (+28)
0x10010000	0x00000014	0x00000000	0x00000004	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010020	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010040	0x00000040	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010060	0x00000060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010080	0x00000080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100a0	0x000000a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100c0	0x000000c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100e0	0x000000e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010100	0x000000f0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010120	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010140	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010160	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010180	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

Registers:

Name	Number	Value
rzero	0	0x00000000
\$t0	1	0x00000000
\$t1	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$a2	8	0x00000000
\$t1	9	0x00000004
\$t2	10	0x00000008
\$t3	11	0x0000000c
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$t0	16	0x00000000
\$a1	17	0x00000000
\$a3	18	0x00000000
\$a4	19	0x00000000
\$a1	20	0x00000000
\$a5	21	0x00000000
\$a6	22	0x00000000
\$a7	23	0x00000000
\$a8	24	0x00000000
\$t9	25	0x00000000
\$a9	26	0x00000000
\$a1	27	0x00000000
\$a9	28	0x00fffffc
\$fp	30	0x00000000
\$ra	31	0x00000000
pc	34	0x00000040
hi	35	0x00000000
lo	36	0x00000000

//programa 13

```

programa12.asm programma13.asm
1 .data
2 A: .word -100
3
4 .text
5 ori $t0, $zero, 1
6
7 lui $t1, 0x1001
8
9 lw $t3, 0($t1)
10
11 srl $t2, $t3, 31
12
13 beq $t0, $t2, negativo
14 j escrita
15
16 negativo:
17 sub $t3, $zero, $t3
18
19 escrita:
20 sw $t3 0($t1)
21

```

Text Segment									Data Segment			
Beg	Address	Code	Basic	Source						Name	Number	Value
	0x00400000	0x40000001	ori \$t0,\$zero,1							zzero	0	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							tat	1	0x00000000
	0x00400000	0x40000000	beq \$t0,\$t2,negativo							z0	2	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							cvt	3	0x00000000
	0x00400000	0x40000000	lw \$t3,0(\$t1)							da0	4	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							ca1	5	0x00000000
	0x00400000	0x40000000	srl \$t2,\$t3,31							ca2	6	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							ca3	7	0x00000000
	0x00400000	0x40000000	beq \$t0,\$t2,negativo							ds0	8	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							ct1	9	0x00000000
	0x00400000	0x40000000	j escrita							ct2	10	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							ct3	11	0x00000000
	0x00400000	0x40000000	sub \$t3,\$zero,\$t3							ct4	12	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							ct5	13	0x00000000
	0x00400000	0x40000000	sw \$t3,0(\$t1)							ct6	14	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							ct7	15	0x00000000
	0x00400000	0x40000000	sub \$t3,\$zero,\$t3							ct8	16	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							ct9	17	0x00000000
	0x00400000	0x40000000	sw \$t3,0(\$t1)							ct10	18	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							ct11	19	0x00000000
	0x00400000	0x40000000	sub \$t3,\$zero,\$t3							ct12	20	0x00000000
	0x00400000	0x3c091001	lui \$t1,0x10000001							ct13	21	0x00000000

//programa 14

//programa 15

```
programa12.asm programa13.asm programa14.asm programa15.asm
1 .text
2 ori $t0, $zero, 100
3 ori $t1, $zero, 0
4 ori $t6, $zero, 1
5 lui $t5, 0x1001
6 add $t7, $zero, $t5
7
8 loop:
9 beq $t0, $zero, parte2
10 sll $t2, $t1, 1
11 add $t2, $t2, $t6
12 sw $t2, 0($t7)
13
14 addi $t7, $t7, 4
15 sub $t0, $t0, $t6
16 add $t1, $t1, $t6
17 j loop
18
19 parte2:
20 ori $t0, $zero, 100
21 ori $t2, $zero, 0
22 soma:
23 beq $t0, $zero, fim
24 lw $t1, 0($t5)
25 add $t2, $t1, $t2
26
27 addi $t5, $t5, 4
28 sub $t0, $t0, $t6
29 j soma
30
31 fim:
32
```

//programa 16

```
programa12.asm programa13.asm programa14.asm programa15.asm
1 .data
2 x:.word 0x186A00
3 y:.word 0x13880
4 z:.word 0x61A80
5
6 .text
7 lui $t0, 0x1001
8
9 lw $s0, 0($t0)      #s0 = x
10 lw $s1, 4($t0)      #s1 = y
11 lw $s2, 8($t0)      #s2 = z
12
13 mult $s0, $s1 # x * y
14 mflo $t1 # resultado da mult
15
16 div $t1, $s2
17 mflo $t2 # resultado da div
```

Registers

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0x00000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x00000000
\$t2	10	0xffffffff
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$a0	16	0x00000000
\$a1	17	0x00000000
\$a2	18	0x00000000
\$a3	19	0x00000000
\$t4	20	0x00000000
\$t5	21	0x00000000
\$t6	22	0x00000000
\$t7	23	0x00000000
\$t8	24	0x00000000
\$t9	25	0x00000000
\$k0	26	0x00000000
\$k1	27	0x00000000
\$sp	28	0xffffffff
\$fp	29	0xffffffff
\$ra	30	0x00000000
pc	31	0x00000000
hi		0xffffffff
lo		0xffffffff

Text Segment

Blgt	Address	Code	Basic	Source
0x04000000	0x00000000	lui \$t0, 0x1001	7: lui \$t0, 0x1001	
0x04000004	0x00000000	lw \$t1, 0(\$t0)	8: lw \$t1, 0(\$t0) # \$t0 = x	
0x04000008	0x00000000	lw \$t2, 4(\$t0)	9: lw \$t2, 4(\$t0) # \$t0 = y	
0x0400000c	0x00000000	li \$t3, 1	10: li \$t3, 1 # t3 = 1	
0x04000010	0x00000000	mult \$t1, \$t2, \$t3	11: mult \$t1, \$t2, \$t3 # resultado da mult	
0x04000014	0x00000000	mflo \$t1	12: mflo \$t1 # resultado da mult	
0x04000018	0x00000000	div \$t1, \$t2	13: div \$t1, \$t2	
0x0400001c	0x00000000	mflo \$t0	14: mflo \$t0 # resultado da div	

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0x00100000	0x00001380	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010020	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010040	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010120	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010140	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010160	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010180	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

Registers

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0x00000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x00000004
\$t2	10	0x00000000
\$t3	11	0x00000001
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$t8	16	0x00000000
\$t9	17	0x00000000
\$k0	18	0x00000000
\$k1	19	0x00000000
\$sp	20	0x00000000
\$fp	21	0xffffffff
\$ra	22	0x00000000
pc	23	0x00000000
hi	24	0xffffffff
lo	25	0xffffffff

//programa 17

programa12.asm

```

1 .data
2 x: .word 10
3 y: .word 200
4
5 .text
6
7 lui $t0, 0x1001
8 lw $t1, 0($t0)
9 lw $t2, 4($t0)
10 ori $t3, $zero, 1
11
12 loop:
13 beq $t2, $zero, fim
14 sub $t2, $t2, $t3
15 add $t4, $t4, $t1
16 j loop
17
18 sw $t4, 8($t0)
19
20

```

Text Segment

Blgt	Address	Code	Basic	Source
0x04000000	0x00000000	lui \$t0, 0x1001	7: lui \$t0, 0x1001	
0x04000004	0x00000000	lw \$t1, 0(\$t0)	8: lw \$t1, 0(\$t0) # \$t0 = x	
0x04000008	0x00000000	lw \$t2, 4(\$t0)	9: lw \$t2, 4(\$t0) # \$t0 = y	
0x0400000c	0x00000000	ori \$t3, \$zero, 1	10: ori \$t3, \$zero, 1	
0x04000010	0x00000000	beq \$t2, \$zero, fim	11: beq \$t2, \$zero, fim	
0x04000014	0x00000000	sub \$t2, \$t2, \$t3	12: sub \$t2, \$t2, \$t3	
0x04000018	0x00000000	add \$t4, \$t4, \$t1	13: add \$t4, \$t4, \$t1	
0x0400001c	0x00000000	j loop	14: j loop	
0x04000020	0x00000000	sw \$t4, 8(\$t0)	15: sw \$t4, 8(\$t0)	

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010020	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010040	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010120	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010140	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010160	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010180	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

Registers

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0x00000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x00000004
\$t2	10	0x00000000
\$t3	11	0x00000001
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$t8	16	0x00000000
\$t9	17	0x00000000
\$k0	18	0x00000000
\$k1	19	0x00000000
\$sp	20	0x00000000
\$fp	21	0xffffffff
\$ra	22	0x00000000
pc	23	0x00000000
hi	24	0xffffffff
lo	25	0xffffffff

//programa 18

```

programa12.asm programa13.asm programa14.asm programa15.asm programa16.asm programa17.asm programa18.asm*
1 .data
2 X: .word 1
3 Y: .word 0
4
5 .text
6
7 # Logica para entender o codigo:
8 # 5^2 = 5 * 5 = (5+5+5+5)
9 # 5^3 = 5 * 5 * 5 = (5+5+5+5) + (5+5+5+5) + (5+5+5+5) + (5+5+5+5)
10 # 5^4 = 5*5*5*5 = 5^3 + 5^3 + 5^3 + 5^3 + 5^3
11
12 lui $t0, 0x1001
13 lw $t1, 0($t0) # t1 = X
14 lw $t2, 4($t0) # t2 = Y
15 ori $t3, $zero, 1 # t3 = 1
16 or $t5, $zero, $t1 # t5 = X
17 or $t7, $zero, $t1 # t7 = X
18 addi $t6, $t0, -1 # t6 = Y - 1
19
20
21 beq $t2, $zero, exp_0 # Caso o expoente for 0, go to exp_0
22 beq $t6, $zero, loop_2 # Caso o expoente for 1, go to loop_2. Porque nao entraria no loop_1, porque t6 = 0
23
24 loop_1:
25 beq $t6, $zero, fim # Repetir (y-1) vezes
26 or $t4, $zero, $zero # t4 = 0 - importante a partir do segundo loop_1
27
28 loop_2:
29 beq $t5, $zero, controle # repetir x vezes e then go to controle
30 sub $t5, $t5, $t3 # t5 = t5 - 1
31 add $t4, $t4, $t1 # t4 = t4 + t1 (resultado)
32 beq $t2, $t3, fim # Se t2 = 1 sair
33 j loop_2
34
35 controle:
36 or $t5, $zero, $t7 # t5 = X
37 or $t1, $zero, $t4 # t1 = t4
38 sub $t6, $t6, $t3 # t6 = t6 - 1
39 j loop_1
40
41
42 exp_0:
43 ori $t4, $zero, 1 # resultado = 1
44
45 fim:
46 sw $t4, 8($t0) # Escrever o resultado
47

```

Edit Execute

Text Segment		Registers Coproc 1 Coproc 0				
Byte	Address	Code	Basic	Name	Number	Value
0x04000000	0x00000000	0xd0000000	0x00000000	\$t0	0	0x00000000
0x04000004	0x00000004	0xd0d00000	0x00000000	\$t1	1	0x00000001
0x04000008	0x00000008	0xd0d00001	0x00000000	\$t2	2	0x00000000
0x0400000c	0x0000000c	0xd0d00002	0x00000000	\$t3	3	0x00000000
0x04000010	0x00000010	0xd0d00003	0x00000000	\$t4	4	0x00000000
0x04000014	0x00000014	0xd0d00004	0x00000000	\$t5	5	0x00000000
0x04000018	0x00000018	0xd0d00005	0x00000000	\$t6	6	0x00000000
0x04000022	0x00000022	0xd0d00006	0x00000000	\$t7	7	0x00000000
0x04000026	0x00000026	0xd0d00007	0x00000000	\$t11	9	0x00000001
0x04000030	0x00000030	0xd0d00008	0x00000000	\$t12	10	0x00000000
0x04000034	0x00000034	0xd0d00009	0x00000000	\$t13	11	0x00000001
0x04000038	0x00000038	0xd0d0000a	0x00000000	\$t14	12	0x00000001
0x04000042	0x00000042	0xd0d0000b	0x00000000	\$t15	13	0x00000001
0x04000046	0x00000046	0xd0d0000c	0x00000000	\$t6	14	0xffffffff
0x04000050	0x00000050	0xd0d0000d	0x00000000	\$t7	15	0x00000001
0x04000054	0x00000054	0xd0d0000e	0x00000000	\$t8	16	0x00000000
0x04000058	0x00000058	0xd0d0000f	0x00000000	\$t9	17	0x00000000
0x04000062	0x00000062	0xd0d00010	0x00000000	\$t2	18	0x00000000
0x04000066	0x00000066	0xd0d00011	0x00000000	\$t3	19	0x00000000
0x04000070	0x00000070	0xd0d00012	0x00000000	\$t4	20	0x00000000
0x04000074	0x00000074	0xd0d00013	0x00000000	\$t5	21	0x00000000
0x04000078	0x00000078	0xd0d00014	0x00000000	\$t6	22	0x00000000
0x04000082	0x00000082	0xd0d00015	0x00000000	\$t7	23	0x00000000
0x04000086	0x00000086	0xd0d00016	0x00000000	\$t8	24	0x00000000
0x04000090	0x00000090	0xd0d00017	0x00000000	\$t9	25	0x00000000
0x04000094	0x00000094	0xd0d00018	0x00000000	\$t10	26	0x00000000
0x04000098	0x00000098	0xd0d00019	0x00000000	\$t11	27	0x00000001
0x040000a2	0x000000a2	0xd0d0001a	0x00000000	\$t12	28	0x00000000
0x040000a6	0x000000a6	0xd0d0001b	0x00000000	\$t13	29	0xffffffff
0x040000b0	0x000000b0	0xd0d0001c	0x00000000	\$t14	30	0x00000000
0x040000b4	0x000000b4	0xd0d0001d	0x00000000	\$t15	31	0x00000000
0x040000b8	0x000000b8	0xd0d0001e	0x00000000	\$t16	32	0x00000000
0x040000c2	0x000000c2	0xd0d0001f	0x00000000	\$t17	33	0x00000000
0x040000c6	0x000000c6	0xd0d00020	0x00000000	\$t18	34	0x00000000
0x040000d0	0x000000d0	0xd0d00021	0x00000000	\$t19	35	0x00000000
0x040000d4	0x000000d4	0xd0d00022	0x00000000	\$t20	36	0x00000000
0x040000d8	0x000000d8	0xd0d00023	0x00000000	\$t21	37	0x00000000
0x040000e2	0x000000e2	0xd0d00024	0x00000000	\$t22	38	0x00000000
0x040000e6	0x000000e6	0xd0d00025	0x00000000	\$t23	39	0x00000000
0x040000f0	0x000000f0	0xd0d00026	0x00000000	\$t24	40	0x00000000
0x040000f4	0x000000f4	0xd0d00027	0x00000000	\$t25	41	0x00000000
0x040000f8	0x000000f8	0xd0d00028	0x00000000	\$t26	42	0x00000000
0x040000fc	0x000000fc	0xd0d00029	0x00000000	\$t27	43	0x00000000
0x04000100	0x00000100	0xd0d0002a	0x00000000	\$t28	44	0x00000000
0x04000104	0x00000104	0xd0d0002b	0x00000000	\$t29	45	0x00000000
0x04000108	0x00000108	0xd0d0002c	0x00000000	\$t30	46	0x00000000
0x04000112	0x00000112	0xd0d0002d	0x00000000	\$t31	47	0x00000000
0x04000116	0x00000116	0xd0d0002e	0x00000000	\$t32	48	0x00000000
0x04000120	0x00000120	0xd0d0002f	0x00000000	\$t33	49	0x00000000
0x04000124	0x00000124	0xd0d00030	0x00000000	\$t34	50	0x00000000
0x04000128	0x00000128	0xd0d00031	0x00000000	\$t35	51	0x00000000
0x04000132	0x00000132	0xd0d00032	0x00000000	\$t36	52	0x00000000
0x04000136	0x00000136	0xd0d00033	0x00000000	\$t37	53	0x00000000
0x04000140	0x00000140	0xd0d00034	0x00000000	\$t38	54	0x00000000
0x04000144	0x00000144	0xd0d00035	0x00000000	\$t39	55	0x00000000
0x04000148	0x00000148	0xd0d00036	0x00000000	\$t40	56	0x00000000
0x04000152	0x00000152	0xd0d00037	0x00000000	\$t41	57	0x00000000
0x04000156	0x00000156	0xd0d00038	0x00000000	\$t42	58	0x00000000
0x04000160	0x00000160	0xd0d00039	0x00000000	\$t43	59	0x00000000
0x04000164	0x00000164	0xd0d0003a	0x00000000	\$t44	60	0x00000000
0x04000168	0x00000168	0xd0d0003b	0x00000000	\$t45	61	0x00000000
0x04000172	0x00000172	0xd0d0003c	0x00000000	\$t46	62	0x00000000
0x04000176	0x00000176	0xd0d0003d	0x00000000	\$t47	63	0x00000000
0x04000180	0x00000180	0xd0d0003e	0x00000000	\$t48	64	0x00000000
0x04000184	0x00000184	0xd0d0003f	0x00000000	\$t49	65	0x00000000
0x04000188	0x00000188	0xd0d00040	0x00000000	\$t50	66	0x00000000
0x04000192	0x00000192	0xd0d00041	0x00000000	\$t51	67	0x00000000
0x04000196	0x00000196	0xd0d00042	0x00000000	\$t52	68	0x00000000
0x040001a0	0x000001a0	0xd0d00043	0x00000000	\$t53	69	0x00000000
0x040001a4	0x000001a4	0xd0d00044	0x00000000	\$t54	70	0x00000000
0x040001a8	0x000001a8	0xd0d00045	0x00000000	\$t55	71	0x00000000
0x040001b2	0x000001b2	0xd0d00046	0x00000000	\$t56	72	0x00000000
0x040001b6	0x000001b6	0xd0d00047	0x00000000	\$t57	73	0x00000000
0x040001b8	0x000001b8	0xd0d00048	0x00000000	\$t58	74	0x00000000
0x040001bc	0x000001bc	0xd0d00049	0x00000000	\$t59	75	0x00000000
0x040001c0	0x000001c0	0xd0d0004a	0x00000000	\$t60	76	0x00000000
0x040001c4	0x000001c4	0xd0d0004b	0x00000000	\$t61	77	0x00000000
0x040001c8	0x000001c8	0xd0d0004c	0x00000000	\$t62	78	0x00000000
0x040001cc	0x000001cc	0xd0d0004d	0x00000000	\$t63	79	0x00000000
0x040001d0	0x000001d0	0xd0d0004e	0x00000000	\$t64	80	0x00000000
0x040001d4	0x000001d4	0xd0d0004f	0x00000000	\$t65	81	0x00000000
0x040001d8	0x000001d8	0xd0d00050	0x00000000	\$t66	82	0x00000000
0x040001dc	0x000001dc	0xd0d00051	0x00000000	\$t67	83	0x00000000
0x040001e0	0x000001e0	0xd0d00052	0x00000000	\$t68	84	0x00000000
0x040001e4	0x000001e4	0xd0d00053	0x00000000	\$t69	85	0x00000000
0x040001e8	0x000001e8	0xd0d00054	0x00000000	\$t70	86	0x00000000
0x040001f0	0x000001f0	0xd0d00055	0x00000000	\$t71	87	0x00000000
0x040001f4	0x000001f4	0xd0d00056	0x00000000	\$t72	88	0x00000000
0x040001f8	0x000001f8	0xd0d00057	0x00000000	\$t73	89	0x00000000
0x040001fc	0x000001fc	0xd0d00058	0x00000000	\$t74	90	0x00000000
0x04000200	0x00000200	0xd0d00059	0x00000000	\$t75	91	0x00000000
0x04000204	0x00000204	0xd0d0005a	0x00000000	\$t76	92	0x00000000
0x04000208	0x00000208	0xd0d0005b	0x00000000	\$t77	93	0x00000000
0x04000212	0x00000212	0xd0d0005c	0x00000000	\$t78	94	0x00000000
0x04000216	0x00000216	0xd0d0005d	0x00000000	\$t79	95	0x00000000
0x04000220	0x00000220	0xd0d0005e	0x00000000	\$t80	96	0x00000000
0x04000224	0x00000224	0xd0d0005f	0x00000000	\$t81	97	0x00000000
0x04000228	0x00000228	0xd0d00060	0x00000000	\$t82	98	0x00000000
0x04000232	0x00000232	0xd0d00061	0x00000000	\$t83	99	0x00000000
0x04000236	0x00000236	0xd0d00062	0x00000000	\$t84	100	0x00000000
0x04000240	0x00000240	0xd0d00063	0x00000000	\$t85	101	0x00000000
0x04000244	0x00000244	0xd0d00064	0x00000000	\$t86	102	0x00000000
0x04000248	0x00000248	0xd0d00065	0x00000000	\$t87	103	0x00000000
0x04000252	0x00000252	0xd0d00066	0x00000000	\$t88	104	0x00000000
0x04000256	0x00000256	0xd0d00067	0x00000000	\$t89	105	0x00000000
0x04000260	0x00000260	0xd0d00068	0x00000000	\$t90	106	0x00000000
0x04000264	0x00000264	0xd0d00069	0x00000000	\$t91	107	0x00000000
0x04000268	0x00000268	0xd0d0006a	0x00000000	\$t92	108	0x00000000
0x04000272	0x00000272	0xd0d0006b	0x00000000	\$t93	109	0x00000000
0x04000276	0x00000276	0xd0d0006c	0x00000000	\$		

Responda

1. Se tivermos 2 inteiros, cada um com 32 bits, quantos bits podemos esperar para o produto?

- A. 16
- B. 32
- C. 64**
- D. 128

2. Quais os registradores que armazenam os resultados na multiplicação?

- A. high e low
- B. hi e lo**
- C. R0 e R1
- D. \$0 e \$1

3. Qual a operação usada para multiplicar inteiros em comp. de dois?

- A. mult**
- B. multu
- C. multi
- D. mutt

4. Qual instrução move os bits menos significativos da multiplicação para o reg. 8?

- A. move \$8,lo
- B. mvlo \$8,lo
- C. mflo \$8**
- D. addu \$8,\$0,lo

5. Se tivermos dois inteiros, cada um com 32 bits, quantos bits deveremos estar preparados para receber no **quociente**?

- A. 16
- B. 32**
- C. 64
- D. 128

6. Após a instrução div, qual registrador possui o quociente?

- A. lo**
- B. hi
- C. high
- D. \$2

7. Qual a inst. Usada para dividir dois inteiros em comp. de dois?

- A. dv**
- B. divide
- C. divu
- D. div

8. Faça um arithmetic shift right de dois no seguinte padrão de bits: 1001 1011

- A. 1110 0110
- B. 0010 0110
- C. 1100 1101**
- D. 0011 0111

9. Qual o efeito de um **arithmetic shift right** de uma posição?

- A. Se o inteiro for unsigned, o shift o divide por 2. Se o inteiro for signed, o shift o divide por 2.
- B. Se o inteiro for unsigned, o shift o divide por 2. Se o inteiro for signed, o shift pode resultar em um valor errado.
- C. Se o inteiro for unsigned, o shift pode ocasionar um valor errado. Se o inteiro for signed, o shift o divide por 2.
- D. O shift multiplica o número por dois.

10. Qual sequencia de instruções avalia $3x+7$, onde x é iniciado no reg. \$8 e o resultado armazenado em \$9?

A.
ori \$3,\$0,3
mult \$8,\$3
mflo \$9

addi \$9,\$9,7

B.

ori \$3,\$0,3
mult \$8,\$3
addi \$9,\$8,7

C.

ori \$3,\$0,3
mult \$8,\$3
mfhi \$9
addi \$9,\$9,7

D.

mult \$8,3
mflo \$9
addi \$9,\$9,7

//programa 19

```
programa12.asm programa13.asm programa14.asm programa15.asm programa16.asm programa17.asm
1 .data
2 x: .word 0xFFFF
3 y: .word 0xFFFF
4
5 .text
6 lui $t0, 0x1001
7
8 lw $s0, 0($t0)
9 lw $s1, 4($t0)
10
11 ori $t6, $zero, 32
12 ori $t7, $zero, 1
13
14 or $t1, $zero, $s0
15 ori $t2, $zero, 31
16
17 or $t4, $zero, $t1
18 loop:
19 bne $t5, $zero, repetir
20 srl $t5, $t4, 31
21 beq $t5, $t7, repetir
22 sll $t4, $t4, 1
23 addi $t6, $t6, -1
24 j loop
25
26 repetir:
27 beq $t1, $s1, final_quantidade_bits_sig
28 or $t0, $zero, $t6
29 or $t1, $zero, $s1
30 or $t4, $zero, $t1
31 or $t5, $zero, $zero
32 ori $t6, $zero, 32
```

```
32 ori $t6, $zero, 32
33
34 j loop
35
36 final_quantidade_bits_sig:
37 or $t1, $zero, $t6
38 ori $t6, $zero, 32
39
40
41 mult $s0, $s1
42
43 add $t4, $t0, $t1
44
45 slt $t4, $t4, $t6
46
47 beq $t4, $zero, maior_32
48 mflo $s2
49 j fim
50
51
52 maior_32:
53 mfhi $s2
54 mflo $s3
55
56 fim:
```

//programa 20

```
1 .data
2 x: .word 3
3 .text
4 lui $t0, 0x1001
5 ori $t1, $zero, 2
6 lw $s0, 0($t0)
7 div $s0, $t1
8 mfhi $s1
9 mult $s0, $s0
10 mflo $s2 # resultado da multi x^2
11 mult $s2, $s0
12 mflo $s3 # resultado da multi x^3
13 mult $s3, $s0
14 mflo $s4 # resultado da multi x^4
15 mult $s4, $s0
16 mflo $s5 # resultado da multi x^5
17 beq $s1, $zero, primeira_parte
18 j segunda_parte
19 primeira_parte:
20 add $t2, $s4, $s3
21 mult $t1, $s2
22 mflo $s6 # 2*x^2
23 sub $t3, $t2, $s6
24 sw $t3, 4($t0)
25 j fim
26 segunda_parte:
27 sub $t4, $s5, $s3
28 addi $s7, $t4, 1
29 sw $s7, 4($t0)
30 fim:
```

//programa 21

```
1 .data
2 x: .word 2
3 .text
4 lui $t0, 0x1001
5 lw $s0, 0($t0) # valor de x
6 mult $s0, $s0
7 mflo $s2 # resultado da multi x2
8 mult $s2, $s0
9 mflo $s3 # resultado da multi x3
10 mult $s3, $s0
11 mflo $s4 # resultado da multi x4
12 mult $s4, $s0
13 mflo $s5 # resultado da multi x5
14 slt $t1, $s0, $zero
15 beq $t1, $zero, maior
16 j menor
17 maior:
18 addi $t2, $s3, 1
19 sw $t2, 4($t0)
20 j fim
21 menor:
22 addi $t3, $s4, 0xFFFFFFFF
23 sw $t3, 4($t0)
24 fim:
```

